Towards Principled Transfer Learning

Christoph Lampert

TASK-CV Workshop at ECCV, Amsterdam
October 9, 2016

Ad: PhD/Postdoc positions at IST Austria, Vienna

IST Austria Graduate School

- 1+3yr PhD program
- join with BSc or MSc
- full scholarships, no fees

ISTFELLOW PostDoc Program

- curiosity-driven research,
- no teaching duties...

Faculty Positions

- (tenure-track) assistant professor
- professor
- opportunities for sabbaticals

Automatic systems that learn and act autonomously

Automatic systems that learn and act autonomously

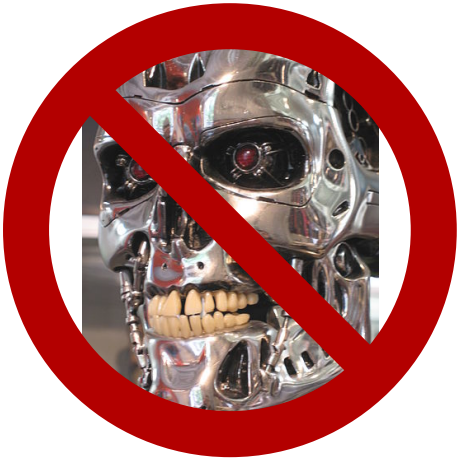


Image: "Terminator's face" by tenaciousme from Tokyo, Japan. Licensed under CC BY 2.0 via Wikimedia Commons
Image: Photo by Basilio Noris

Automatic systems that can analyze and interpret data

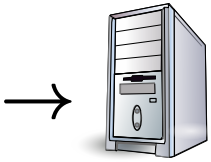
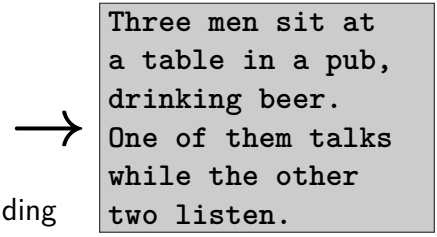
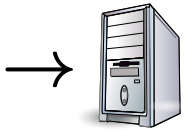


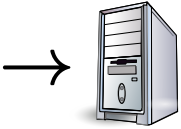
Image Understanding





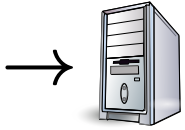
Scene Classification

- outdoors: yes
- in a pub: yes



Action Classification

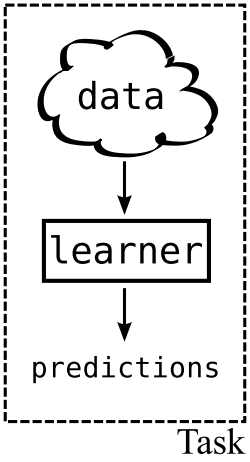
- drinking: yes
- talking: yes

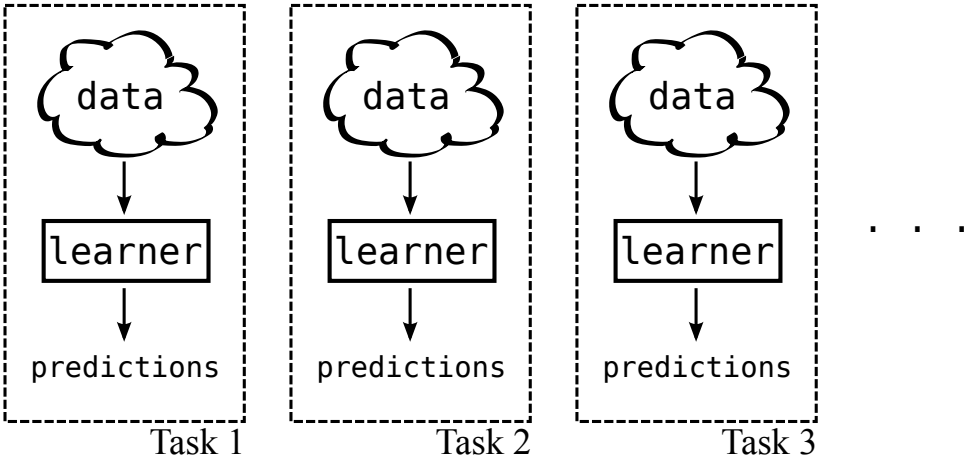


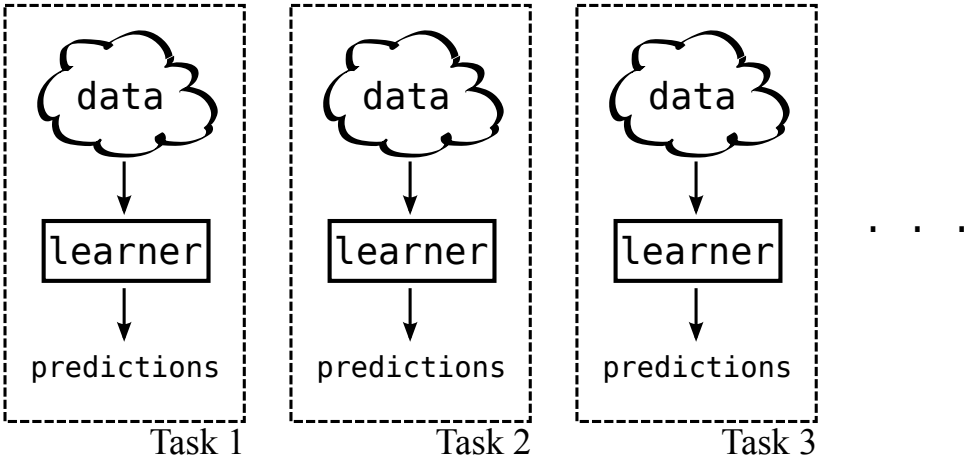
Object Recognition

- persons: 3
- tables: 1
- unicorns: 0

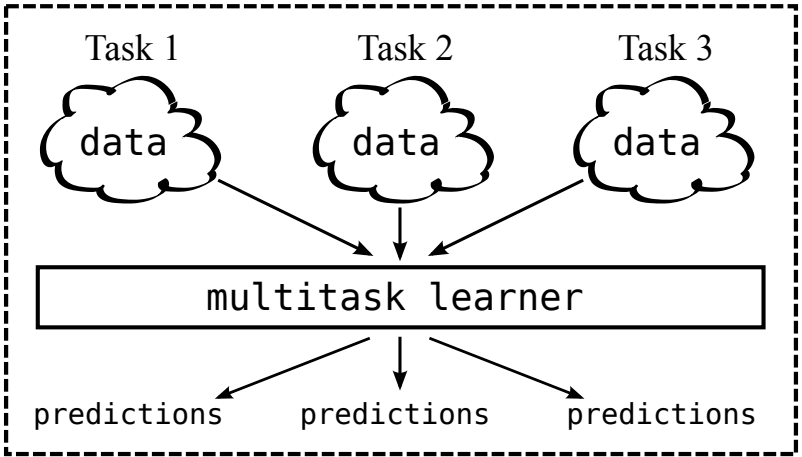
High level view: Learning A Task



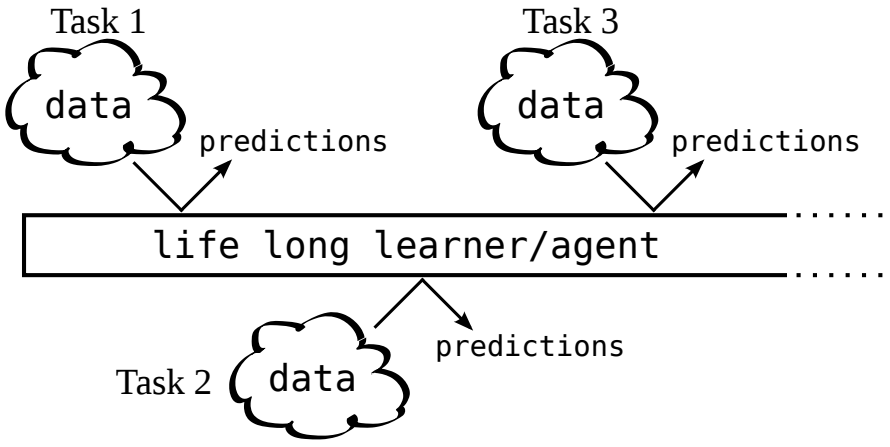

data
learner
predictions
Task 1
data
learner
predictions
Task 2
data
learner
predictions
Task 3
. . .



Tabula Rasa Learning?



Multitask Learning



Lifelong Learning

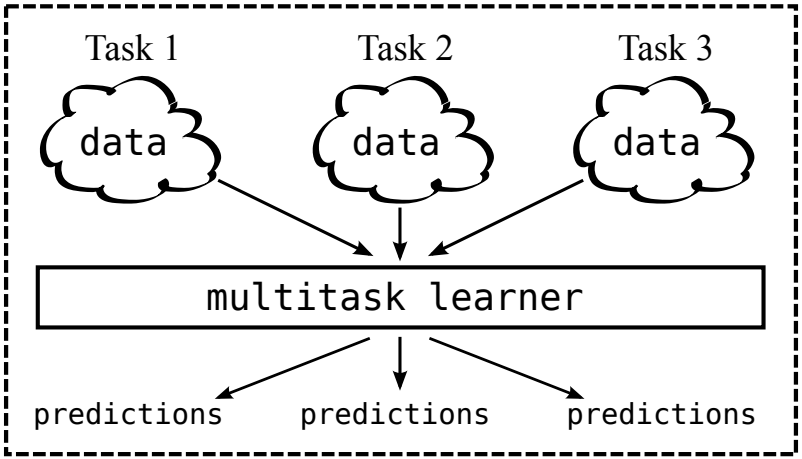
Challenges

- derive methods with guaranteed performance
  - transfer learning trails far behind regular supervised learning
- identify new interesting learning scenarios
  - not just solving the same academic datasets better and better
- find algorithms that are practically useful
  - they must be simple and they must actually work

Active Task Selection for Multi-Task Learning

Asya Pentina

A. Pentina, CHL, *"Active Task Selection for Multi-Task Learning"*, arXiv:1602.06518 [stat.ML]


Task 1
Task 2
Task 3
data
data
data
multitask learner
predictions
predictions
predictions

Example: Personalized Speech Recognition

each user speaks differently: learn an individual classifier

Example: Personalized Spam Filters

each user has different preferences: learn individual filter rules

New setting: multi-task learning with unlabeled tasks

- Given: T tasks
  - for a subset of them labeled training data is available
  - for the others (the majority) only unlabeled samples are available
- Goal: classifiers for all tasks, labeled and unlabeled

New setting: multi-task learning with unlabeled tasks

- Given: T tasks
  - for a subset of them labeled training data is available
  - for the others (the majority) only unlabeled samples are available
- Goal: classifiers for all tasks, labeled and unlabeled

Example 1: speech recognition, $T =$ millions

- each user speaks differently $\rightarrow$ separate classifiers
- collecting some unlabeled data (=speech) from each user is easy
- few users are willing to provide annotation (=go through a training session)

New setting: multi-task learning with unlabeled tasks

- Given: T tasks
  - for a subset of them labeled training data is available
  - for the others (the majority) only unlabeled samples are available
- Goal: classifiers for all tasks, labeled and unlabeled

Example 1: speech recognition, $T =$ millions

- each user speaks differently $\rightarrow$ separate classifiers
- collecting some unlabeled data (=speech) from each user is easy
- few users are willing to provide annotation (=go through a training session)

Example 2: product reviews, $T =$ millions

- different attributes matters for different products $\rightarrow$ separate classifiers
- millions of product reviews are readily available
- for many questions of interest, we don't have numeric scores $\rightarrow$ no labels

Setting: multi-task learning with unlabeled tasks

Strategy 1: single-task approach

- learn a single classifier on training data of all tasks together
- use this classifier for all tasks, labeled and unlabeled

Setting: multi-task learning with unlabeled tasks

Strategy 1: single-task approach

- learn a single classifier on training data of all tasks together
- use this classifier for all tasks, labeled and unlabeled

Strategy 2: multi-task approach

- learn separate classifiers for each labeled task
- form combined classifier and use for all unlabeled tasks

Setting: multi-task learning with unlabeled tasks

Strategy 1: single-task approach

- learn a single classifier on training data of all tasks together
- use this classifier for all tasks, labeled and unlabeled

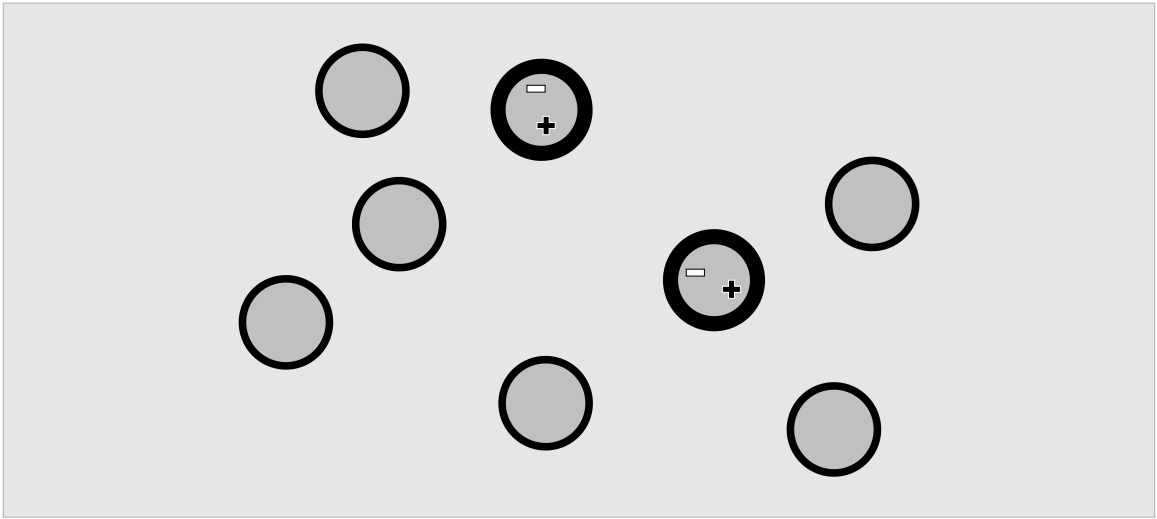
Strategy 2: multi-task approach

- learn separate classifiers for each labeled task
- form combined classifier and use for all unlabeled tasks

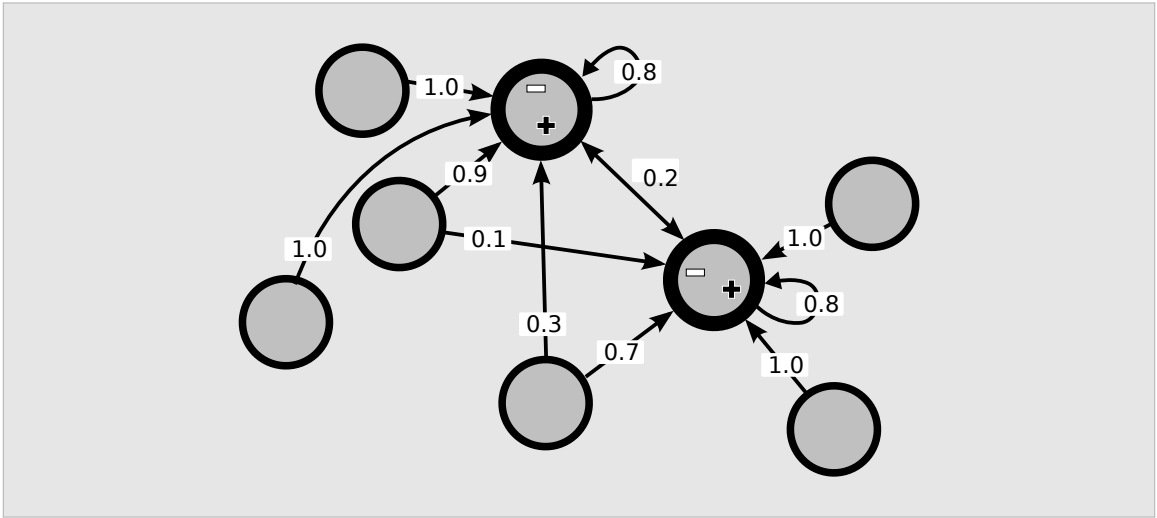
Strategy 3: domain adaptation approach

- for each tasks, learn a classifier on a (weighted) subset of labeled samples

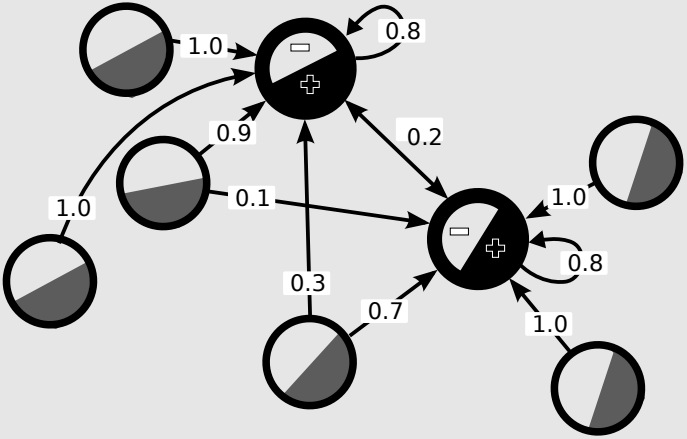
Multi-Task Learning with Unlabeled Tasks



Given: data for multiple tasks (circles), a subset with labels (bold)



Step 1: determine for each tasks from which and how much to share



Step 2: use weighted labeled data to learn classifier for each tasks

Open questions:

- how exactly to do the steps?
- is it going to work?
- under which assumptions?
- can we extend this idea to more general situations?

Our approach:

- prove learning guarantees
- derive a principled algorithm
- confirm experimentally

Reminder: Learning a Task

One learning tasks, $\mathcal{T}$, consists of

- input space, $\mathcal{X}$ (e.g. images or audio samples)
- output space, $\mathcal{Y}$ here: $\mathcal{Y} = \{0, 1\}$
- loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, here: 0/1-loss $\ell(y, \bar{y}) = [\![y \neq \bar{y}]\!]$
- for simplicity: deterministic labels
  - x distributed according to an (unknown) data distribution $D(x)$
  - (unknown) deterministic labeling function $f : \mathcal{X} \to \mathcal{Y}$

Learning scenario

- hypothesis class, $\mathcal{H}$ (e.g. linear classifiers, or deep networks)
- goal: find hypothesis $h \in \mathcal{H}$ with small generalization error

$$\text{er}(h) := \mathbb{E}_{x \sim D}[\, \ell(\, f(x), h(x)\,)\,]$$

Learning Classifiers from a Labeled Training Set

Given a training set $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ for a task, we can compute the training error

$$\widehat{\mathsf{er}}(h) := \frac{1}{n} \sum_{i=1}^{n} \ell(\, y_i, h(x_i)\,).$$

We learn a classifier (e.g. deep network, support vector machines, random forest, ...) by minimizing the training error, potentially with regularization.

Learning Classifiers from a Labeled Training Set

Given a training set $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ for a task, we can compute the training error

$$\widehat{\mathsf{er}}(h) := \frac{1}{n} \sum_{i=1}^{n} \ell(\, y_i, h(x_i)\,).$$

We learn a classifier (e.g. deep network, support vector machines, random forest, ...) by minimizing the training error, potentially with regularization.

Goal of learning theory:

- connect *training error* $\widehat{\mathsf{er}}(h)$ and *generalization error* $\mathsf{er}(h)$

Multi-task learning with unlabeled tasks

Situation:

- $\langle D_1, f_1 \rangle, \ldots, \langle D_T, f_T \rangle$: tasks
- $S_1, \ldots, S_T$: unlabeled sample sets for T tasks, $|S_t| = n$
- $\overline{S}_1, \ldots, \overline{S}_k$: labeled sample sets for tasks $1, \ldots, k$, $|\overline{S}_t| = m$

General idea:

- for each task, learn a classifier on weighted combination of the training sets of "similar" labeled tasks

Parameterization: weight matrix $\alpha \in \mathbb{R}^{k \times T}$

- α_s^t indicates how much tasks t relies on samples from tasks s
- columns of α sum to 1

Theorem (variant of [Pentina, CHL. 2016])

Let $\delta > 0$. Provided that the choice of weights $\alpha \in \mathbb{R}^{k \times T}$ are determined by the unlabeled data only, then with probability at least $1-\delta$ the following inequality holds for all possible choices of α and $h_1, \ldots, h_T \in \mathcal{H}$:

$$\sum_{t=1}^{T} \mathrm{er}_t(h_t) \leq \sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \widehat{\mathrm{er}}_i(h_t) + \sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \,\mathrm{disc}(S_t, S_i) + \sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \lambda_{ti} + A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2},$$

$$+ \text{some } O(T/\sqrt{n}) \text{ and } O(T/\sqrt{m}) \text{ complexity terms}$$

$$\|\alpha\|_{2,1} = \sum_{t=1}^{T} \sqrt{\sum_{i=1}^{k} (\alpha_i^t)^2}, \quad \|\alpha\|_{1,2} = \sqrt{\sum_{i=1}^{k} \left(\sum_{t=1}^{T} \alpha_i^t\right)^2}, \quad A = \sqrt{\frac{2d\log(ekm/d)}{m}}, \quad B = \sqrt{\frac{\log(4/\delta)}{2m}}.$$

Theorem (variant of [Pentina, CHL. 2016])

Let $\delta > 0$. Provided that the choice of weights $\alpha \in \mathbb{R}^{k \times T}$ are determined by the unlabeled data only, then with probability at least $1-\delta$ the following inequality holds for all possible choices of α and $h_1, \ldots, h_T \in \mathcal{H}$:

$$\sum_{t=1}^{T} \mathsf{er}_t(h_t) \leq \sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \widehat{\mathsf{er}}_i(h_t) + \sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \operatorname{disc}(S_t, S_i) + \sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \lambda_{ti} + A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2},$$

$$+ \text{some } O(T/\sqrt{n}) \text{ and } O(T/\sqrt{m}) \text{ complexity terms}$$

$$\|\alpha\|_{2,1} = \sum_{t=1}^{T} \sqrt{\sum_{i=1}^{k} (\alpha_i^t)^2}, \quad \|\alpha\|_{1,2} = \sqrt{\sum_{i=1}^{k} \left(\sum_{t=1}^{T} \alpha_i^t \right)^2}, \quad A = \sqrt{\frac{2d \log(ekm/d)}{m}}, \quad B = \sqrt{\frac{\log(4/\delta)}{2m}}.$$

As we'll see, this theorem yields:

- insight which similarity measure to use between (unlabeled) tasks
- a principled learning algorithm (i.e. how to choose α and h_t)
- insight which conditions should be fulfilled for the algorithm to work

$$\overbrace{\sum_{t=1}^{T} \mathsf{er}_t(h_t)}^{\text{total generalization error}} \leq \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \widehat{\mathsf{er}}_i(h_t)}^{\text{error on } \alpha\text{-weighted training sets}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \operatorname{disc}(S_t, S_i)}^{\alpha\text{-weighted similarity}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \lambda_{ti}}^{\text{label function agreements}} + \overbrace{A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}}^{\text{regularizers for task selection}} + \ldots$$

$$\overbrace{\sum_{t=1}^{T} \mathsf{er}_t(h_t)}^{\text{total generalization error}} \leq \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \widehat{\mathsf{er}}_i(h_t)}^{\text{error on } \alpha\text{-weighted training sets}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \, \mathrm{disc}(S_t, S_i)}^{\alpha\text{-weighted similarity}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \lambda_{ti}}^{\text{label function agreements}} + \overbrace{A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}}^{\text{regularizers for task selection}} + \ldots$$

Insight: similarity between unlabeled tasks

Discrepancy Distance (here: for finite sample sets) [Mansour *et al.* 2009]

$$\mathrm{disc}(S_1, S_2) = 2 \max_{h,h' \in \mathcal{H}} \left| \frac{1}{|S_1|} \sum_{x \in S_1} [\![h(x) \neq h'(x)]\!] - \frac{1}{|S_2|} \sum_{x \in S_2} [\![h(x) \neq h'(x)]\!] \right|.$$

$$\overbrace{\sum_{t=1}^{T} \mathsf{er}_t(h_t)}^{\text{total generalization error}} \leq \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \widehat{\mathsf{er}}_i(h_t)}^{\text{error on } \alpha\text{-weighted training sets}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \operatorname{disc}(S_t, S_i)}^{\alpha\text{-weighted similarity}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \lambda_{ti}}^{\text{label function agreements}} + \overbrace{A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}}^{\text{regularizers for task selection}} + \ldots$$

Insight: similarity between unlabeled tasks

Discrepancy Distance (here: for finite sample sets) [Mansour *et al*. 2009]

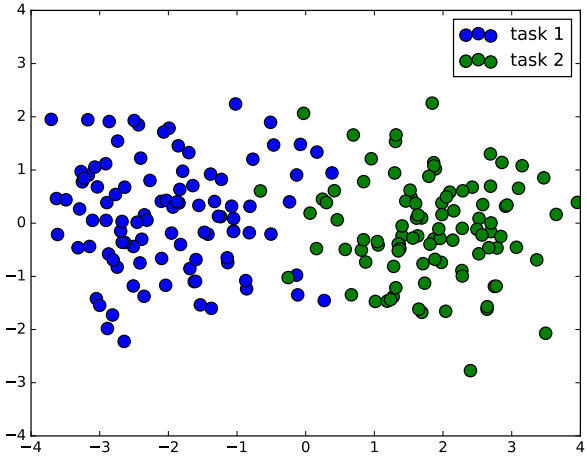
$$\operatorname{disc}(S_1, S_2) = 2 \max_{h,h' \in \mathcal{H}} \left| \frac{1}{|S_1|} \sum_{x \in S_1} [\![h(x) \neq h'(x)]\!] - \frac{1}{|S_2|} \sum_{x \in S_2} [\![h(x) \neq h'(x)]\!] \right|.$$

Lemma [Ben-David *et al*. 2010]

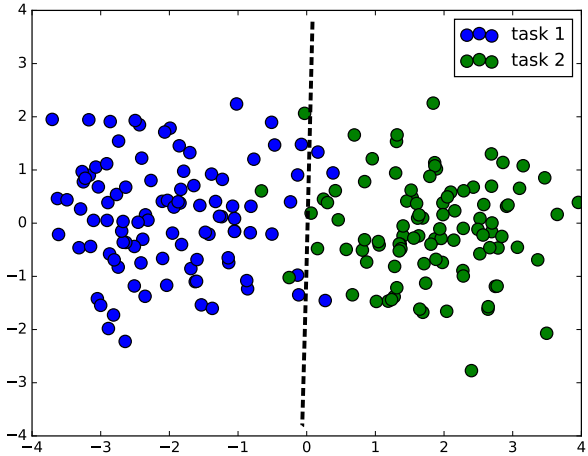
$$\operatorname{disc}(S_1, S_2) = 2(1 - 2e)$$

where e is the training error of the best classifier in $\mathcal{H}$ for the binary classification problem with S_1 as class 1 and S_2 as class 2.

Discrepancy illustration

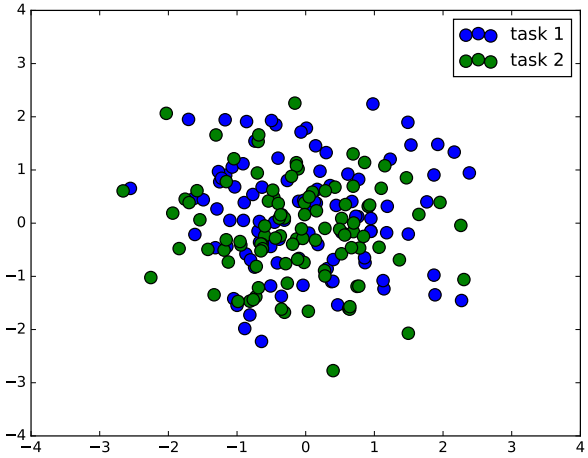


Discrepancy illustration

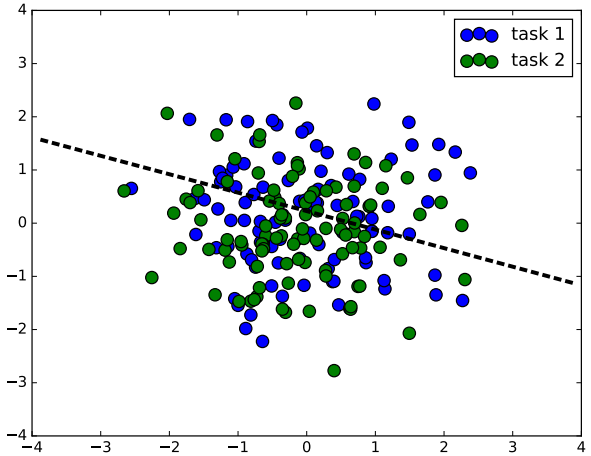


Good separating linear classifier $\rightarrow$ large discrepancy (w.r.t. linear hypotheses class)

Discrepancy illustration

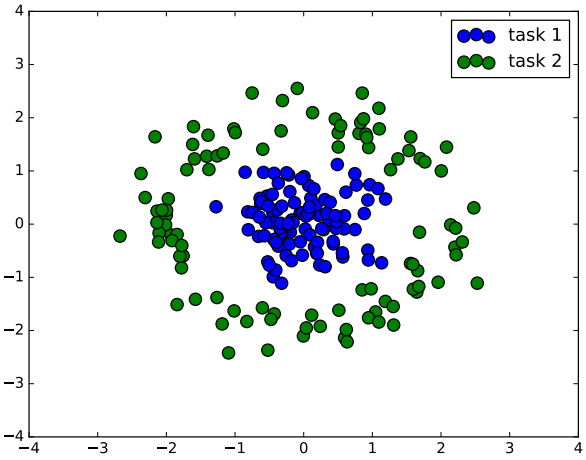


Discrepancy illustration

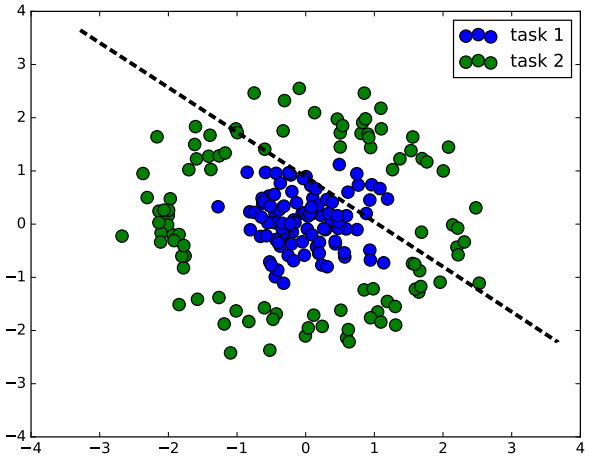


No good separating linear classifier $\rightarrow$ small discrepancy (w.r.t. linear hypotheses class)

Discrepancy illustration

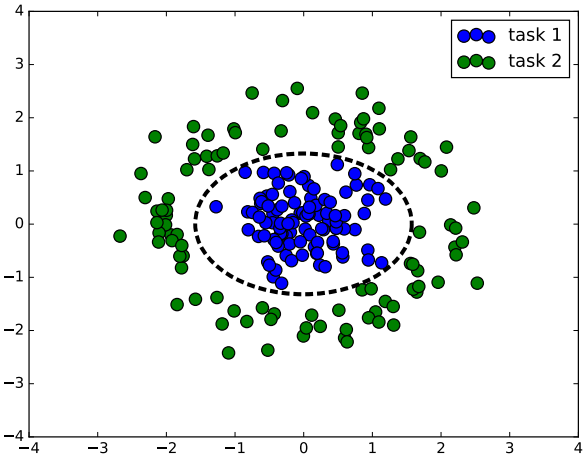


Discrepancy illustration



No good separating linear classifier $\rightarrow$ small discrepancy (w.r.t. linear hypotheses class)

Discrepancy illustration



No good separating linear classifier $\rightarrow$ small discrepancy (w.r.t. linear hypotheses class) but could be large for non-linear hypotheses, e.g., quadratic

$$\overbrace{\sum_{t=1}^{T} \mathsf{er}_t(h_t)}^{\text{total generalization error}} \quad \leq \quad \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \widehat{\mathsf{er}}_i(h_t)}^{\text{error on } \alpha\text{-weighted training sets}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \, \mathrm{disc}(S_t, S_i)}^{\alpha\text{-weighted similarity}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \lambda_{ti}}^{\text{label function agreements}} \quad \overbrace{+ A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2} + \ldots}^{\text{regularizers for task selection}}$$

$$\overbrace{\sum_{t=1}^{T} \mathsf{er}_t(h_t)}^{\text{total generalization error}} \leq \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \widehat{\mathsf{er}}_i(h_t)}^{\text{error on } \alpha\text{-weighted training sets}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \operatorname{disc}(S_t, S_i)}^{\alpha\text{-weighted similarity}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \lambda_{ti}}^{\text{label function agreements}} + \overbrace{A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}}^{\text{regularizers for task selection}} + \ldots$$

Insight: principled learning algorithm

- inequality holds for every choice of α and $h_1, \ldots, h_T$
- most promising choice: use α and $h_1, \ldots, h_T$ that minimize right-hand side

$$\overbrace{\sum_{t=1}^{T} \mathrm{er}_t(h_t)}^{\text{total generalization error}} \leq \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \widehat{\mathrm{er}}_i(h_t)}^{\text{error on } \alpha\text{-weighted training sets}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \,\mathrm{disc}(S_t, S_i)}^{\alpha\text{-weighted similarity}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \lambda_{ti}}^{\text{label function agreements}} + \overbrace{A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}}^{\text{regularizers for task selection}} + \ldots$$

Insight: principled learning algorithm

- inequality holds for every choice of α and $h_1, \ldots, h_T$
- most promising choice: use α and $h_1, \ldots, h_T$ that minimize right-hand side

Algorithm:

- estimate $\mathrm{disc}(S_i, S_j)$ between tasks
- find weights α by minimizing α-weighted discrepancies $+ A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}$
  - large weights between similar tasks, small weights between dissimilar tasks
  - regularizers: spread weights over multiple tasks if possible
- learn classifiers h_t for each task from α^t-weighted sample sets

$$\overbrace{\sum_{t=1}^{T} \mathrm{er}_t(h_t)}^{\text{total generalization error}} \leq \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \widehat{\mathrm{er}}_i(h_t)}^{\text{error on } \alpha\text{-weighted training sets}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \,\mathrm{disc}(S_t, S_i)}^{\alpha\text{-weighted similarity}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \lambda_{ti}}^{\text{label function agreements}} + \overbrace{A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}}^{\text{regularizers for task selection}} + \ldots$$

Insight: principled learning algorithm

- inequality holds for every choice of α and $h_1, \ldots, h_T$
- most promising choice: use α and $h_1, \ldots, h_T$ that minimize right-hand side

Algorithm:

- estimate $\mathrm{disc}(S_i, S_j)$ between tasks
- find weights α by minimizing α-weighted discrepancies $+ A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}$
  - large weights between similar tasks, small weights between dissimilar tasks
  - regularizers: spread weights over multiple tasks if possible
- learn classifiers h_t for each task from α^t-weighted sample sets

$$\overbrace{\sum_{t=1}^{T} \mathrm{er}_t(h_t)}^{\text{total generalization error}} \leq \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \widehat{\mathrm{er}}_i(h_t)}^{\text{error on } \alpha\text{-weighted training sets}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \,\mathrm{disc}(S_t, S_i)}^{\alpha\text{-weighted similarity}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \lambda_{ti}}^{\text{label function agreements}} + \overbrace{A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}}^{\text{regularizers for task selection}} + \ldots$$

Insight: principled learning algorithm

- inequality holds for every choice of α and $h_1, \ldots, h_T$
- most promising choice: use α and $h_1, \ldots, h_T$ that minimize right-hand side

Algorithm:

- estimate $\mathrm{disc}(S_i, S_j)$ between tasks
- find weights α by minimizing α-weighted discrepancies $+ A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}$
  - large weights between similar tasks, small weights between dissimilar tasks
  - regularizers: spread weights over multiple tasks if possible
- learn classifiers h_t for each task from α^t-weighted sample sets

$$\overbrace{\sum_{t=1}^{T} \mathsf{er}_t(h_t)}^{\text{total generalization error}} \leq \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \widehat{\mathsf{er}}_i(h_t)}^{\text{error on } \alpha\text{-weighted training sets}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \operatorname{disc}(S_t, S_i)}^{\alpha\text{-weighted similarity}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \lambda_{ti}}^{\text{label function agreements}} + \overbrace{A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}}^{\text{regularizers for task selection}} + \ldots$$

$$\overbrace{\sum_{t=1}^{T} \mathsf{er}_t(h_t)}^{\text{total generalization error}} \leq \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \widehat{\mathsf{er}}_i(h_t)}^{\text{error on } \alpha\text{-weighted training sets}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \,\mathrm{disc}(S_t, S_i)}^{\alpha\text{-weighted similarity}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \lambda_{ti}}^{\text{label function agreements}} + \overbrace{A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2} + \dots}^{\text{regularizers for task selection}}$$

Insight: limitations/necessary assumptions

$$\lambda_{st} = \min_{h \in H} (\mathsf{er}_s(h) + \mathsf{er}_t(h))$$ *"Is there a classifier that works on both tasks?"*

cannot be computed/estimated without label information.

$$\overbrace{\sum_{t=1}^{T} \mathsf{er}_t(h_t)}^{\text{total generalization error}} \leq \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \widehat{\mathsf{er}}_i(h_t)}^{\text{error on } \alpha\text{-weighted training sets}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \,\mathrm{disc}(S_t, S_i)}^{\alpha\text{-weighted similarity}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \lambda_{ti}}^{\text{label function agreements}} + \overbrace{A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}}^{\text{regularizers for task selection}} + \dots$$

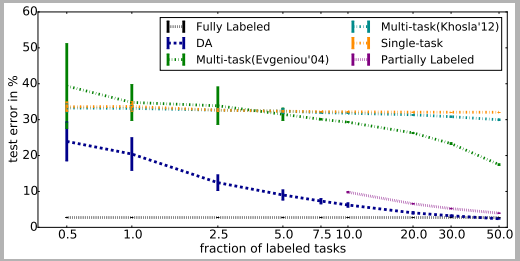
Insight: limitations/necessary assumptions

$$\lambda_{st} = \min_{h \in H} (\mathsf{er}_s(h) + \mathsf{er}_t(h)) \qquad \textit{"Is there a classifier that works on both tasks?"}$$

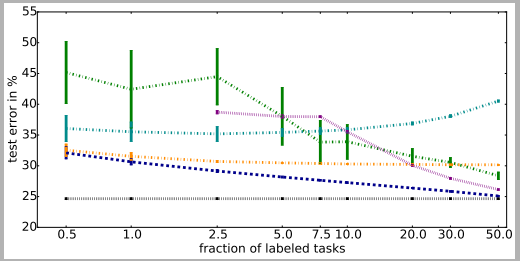
cannot be computed/estimated without label information.

- Bound is informative only if $\sum_{t=1}^{T}\sum_{i=1}^{k} \alpha_i^t \lambda_{ti}$ is small.
- We cannot enforce that. Under which assumptions will it be the case?
- We expect α_i^t to be non-zero only between "similar" tasks
- Algorithm will work if "similar tasks have similar labeling functions"
  - e.g. speech recognition, but not multi-label classification

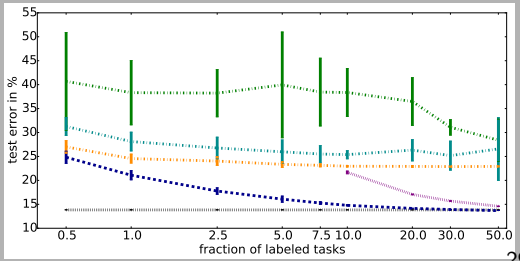
Synthetic (1000 tasks)



Amazon products (957 tasks)



ImageNet 1-vs-others (999 tasks)



- **Fully Labeled**: all T tasks fully labeled (hypothetical)
- **DA**: proposed transfer based on domain adaptation
- **Multi-task**: Multi-task methods with central classifier
- **Single-task**: Single classifier learned from all data
- **Partially Labeled**: all tasks partially labeled (hypo.)

Extension:

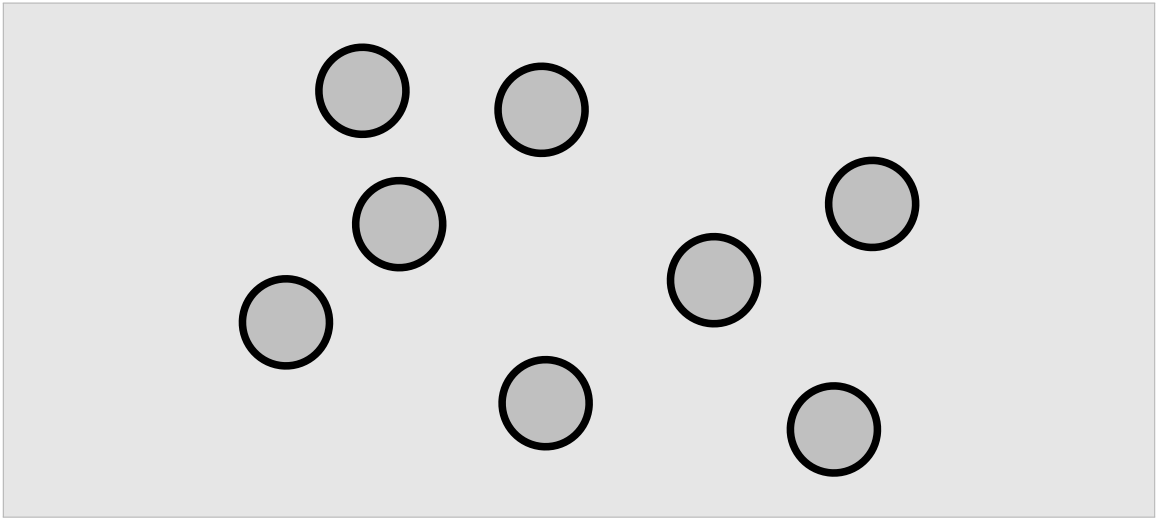
Active Selection of Tasks

New Setting: all task unlabeled

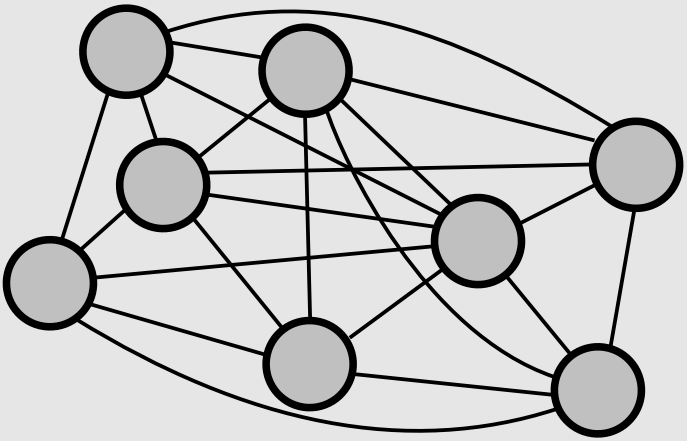
- $\langle D_1, f_1 \rangle, \ldots, \langle D_T, f_T \rangle$: tasks
- $S_1, \ldots, S_T$: unlabeled sample sets for T tasks
- the agent can ask for labels for k of the tasks

Goal:

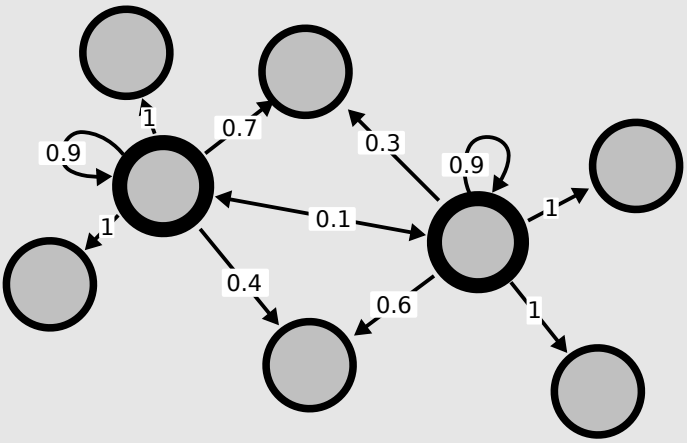
- identify a subset of k tasks to be labeled
- decide between which tasks to share classifiers
- ask for labeled training sets
- learn all jointly classifiers



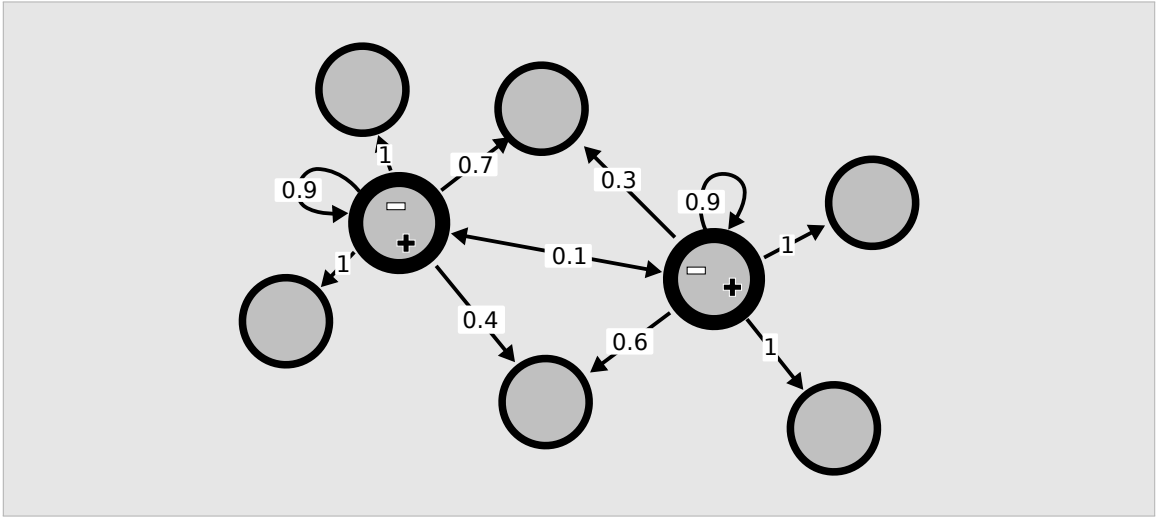
Given: tasks (circles) with only unlabeled data



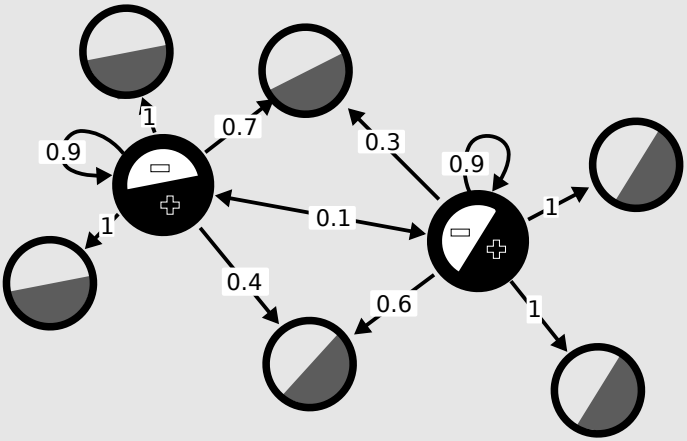
Step 1: compute pairwise discrepancies



Step 2: select prototypes and compute amount of sharing (minimize w.r.t. α)



Step 3: request labels for selected tasks



Step 4: learn classifier for each task using weighted sample sets

Theorem still holds (with different parameterization, $\alpha \in \mathbb{R}^{T \times T}$):

$$\overbrace{\sum_{t=1}^{T} \mathsf{er}_t(h_t)}^{\text{total generalization error}} \leq \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{T} \alpha_i^t \widehat{\mathsf{er}}_i(h_t)}^{\text{error on } \alpha\text{-weighted training sets}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{T} \alpha_i^t \operatorname{disc}(S_t, S_i)}^{\alpha\text{-weighted similarity}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{T} \alpha_i^t \lambda_{ti}}^{\text{label function agreements}} + \overbrace{A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}}^{\text{regularizers for task selection}} + \ldots$$

Strategy: minimize (computable terms of) the right hand side over α and $h_1, \ldots, h_T$

Theorem still holds (with different parameterization, $\alpha \in \mathbb{R}^{T \times T}$):

$$\overbrace{\sum_{t=1}^{T} \mathsf{er}_t(h_t)}^{\text{total generalization error}} \leq \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{T} \alpha_i^t \widehat{\mathsf{er}}_i(h_t)}^{\text{error on } \alpha\text{-weighted training sets}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{T} \alpha_i^t \operatorname{disc}(S_t, S_i)}^{\alpha\text{-weighted similarity}} + \overbrace{\sum_{t=1}^{T}\sum_{i=1}^{T} \alpha_i^t \lambda_{ti}}^{\text{label function agreements}} + \overbrace{A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}}^{\text{regularizers for task selection}} + \ldots$$

Strategy: minimize (computable terms of) the right hand side over α and $h_1, \ldots, h_T$

Algorithm:

- estimate $\operatorname{disc}(S_i, S_j)$ between all tasks
- minimize α-weighted discrepancies + $A\|\alpha\|_{2,1} + B\|\alpha\|_{1,2}$ w.r.t. α
- request labels for tasks corresponding to non-zero rows of α
- learn classifiers h_t for each task from α^t-weighted sample sets

Simplification:

Single-Source Transfer

Can we make this even simpler and more efficient?

So far:

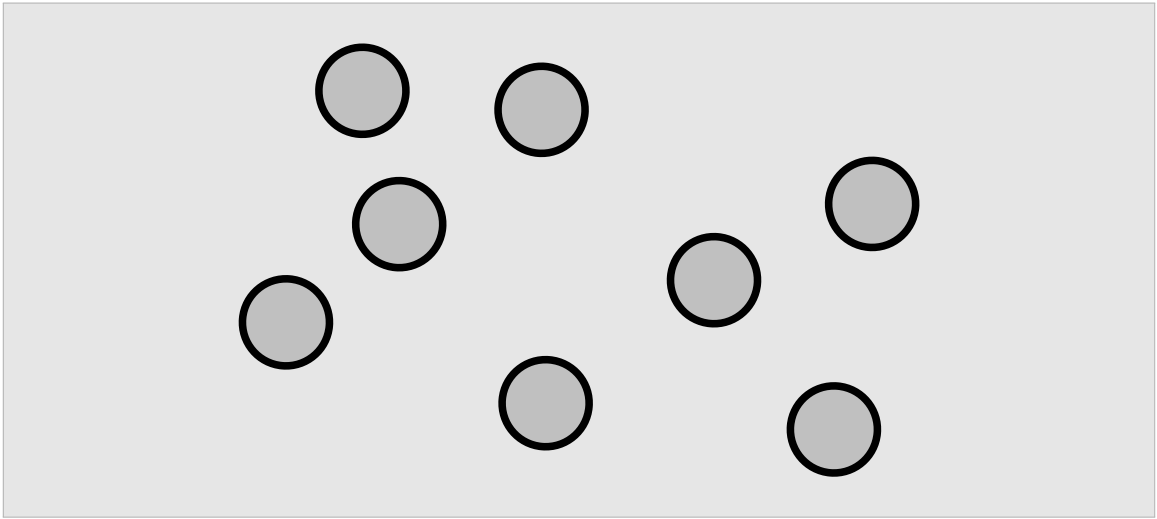
- we learn T classifiers
- each one uses up to mk (weighted) data points

Suggestion: learn classifiers only for labeled tasks and re-use them for unlabeled ones

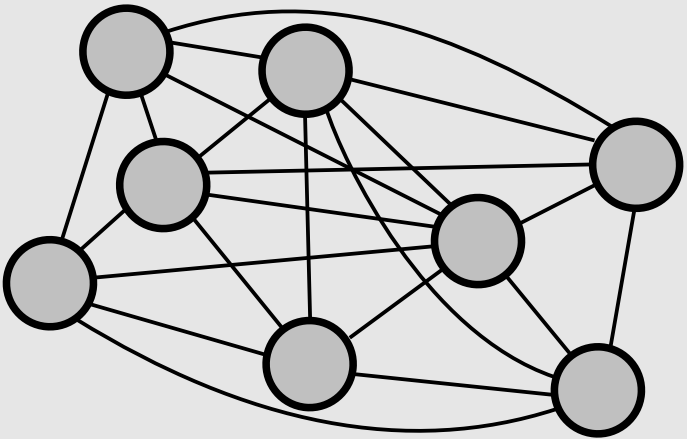
- we learn k classifiers
- each one uses m data points

Corresponds to just additional constraints on $\alpha \rightarrow$ theorem still holds

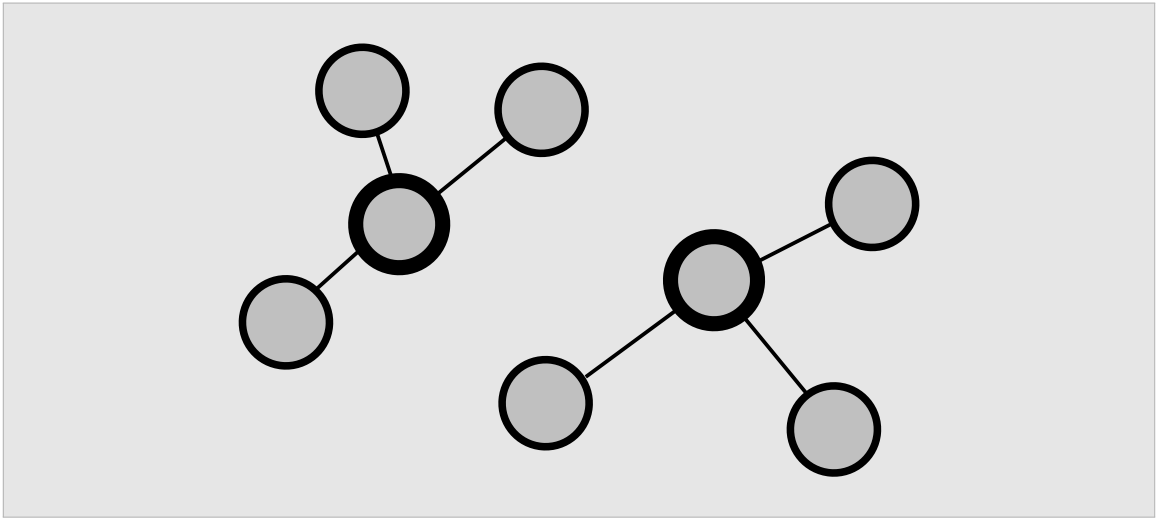
Active Task Selection for Multi-Task Learning: single-source transfer



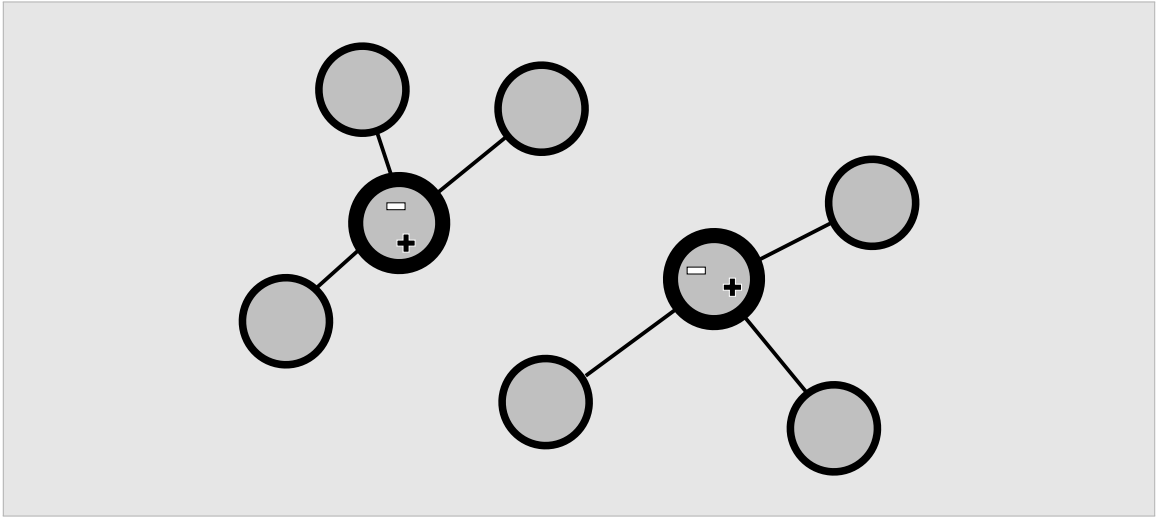
Given: tasks (circles) with only unlabeled data



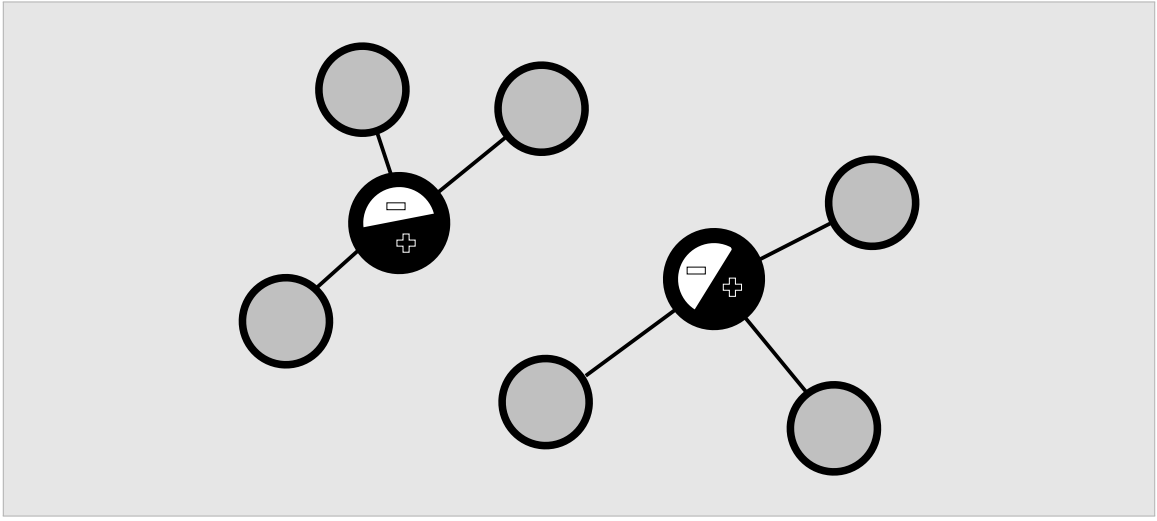
Step 1: compute pairwise discrepancies



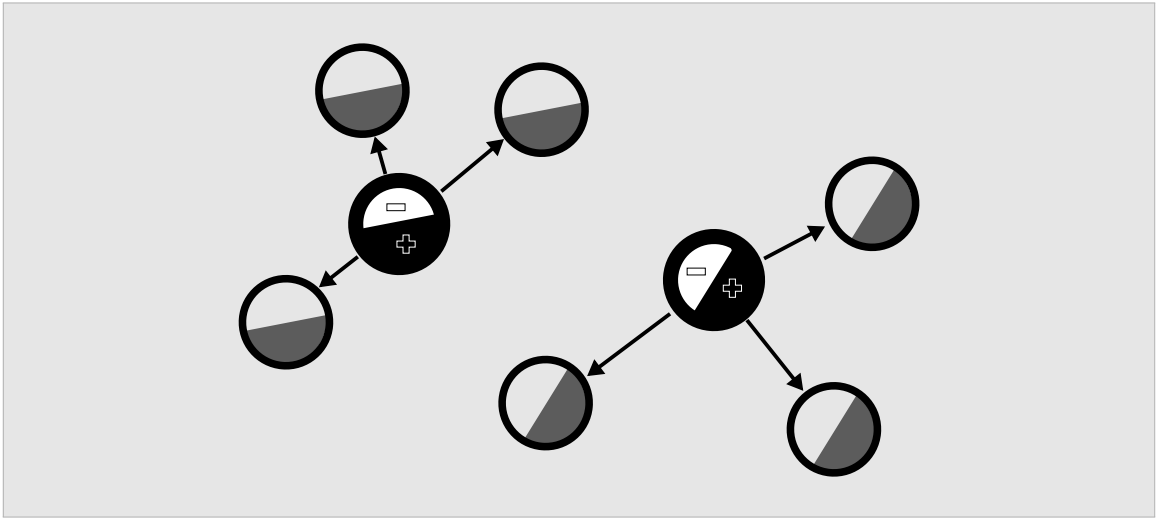
Step 2: minimize bound w.r.t. assignments **(=k-mediod clustering)**



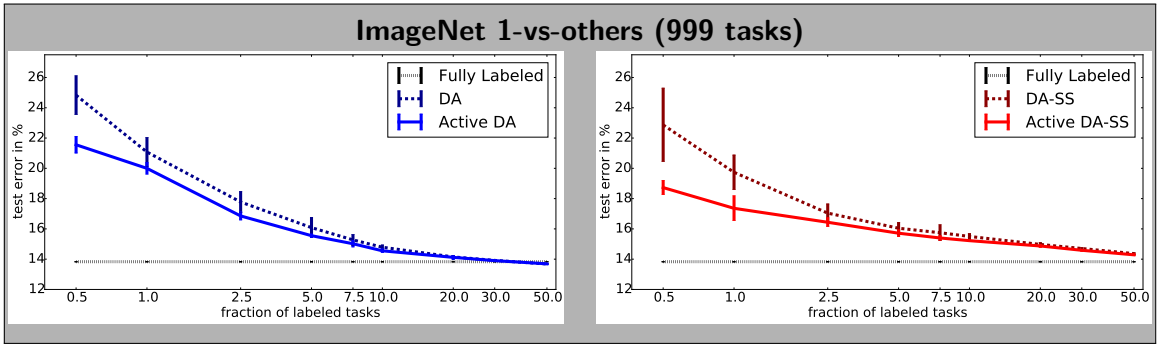
Step 3: request labels for tasks corresponding to cluster centroids



Step 4: learn classifiers for centroid tasks

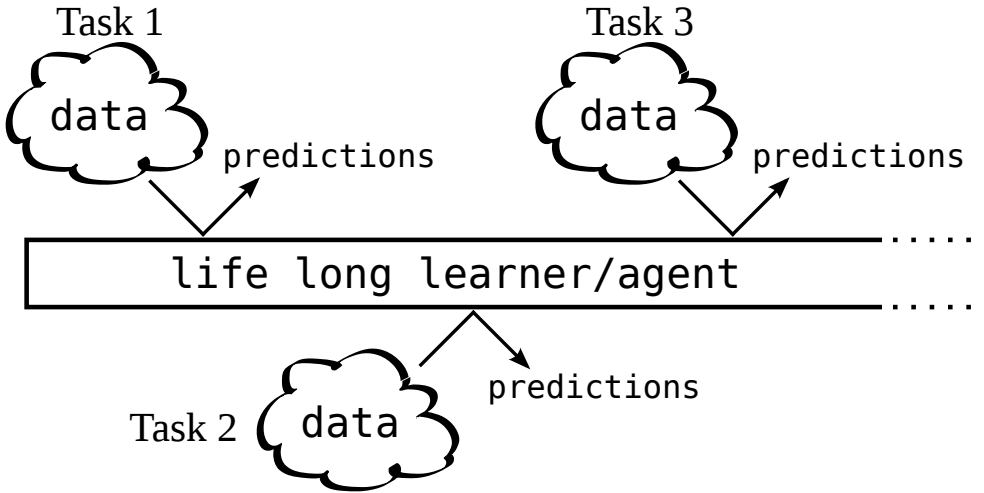


Step 5: transfer classifiers to other tasks in cluster



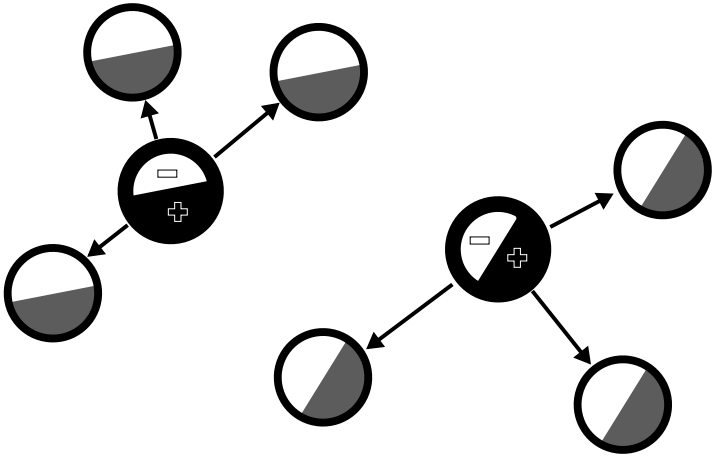
- **Fully Labeled**: all T tasks labeled (hypothetical baselines)
- **DA**: random labeled tasks, discrepancy-based transfer
- **Active DA**: active selection of labeled tasks, discrepancy-based transfer
- **DA-SS**: fixed labeled tasks, discrepancy-based transfer (single source)
- **Active DA-SS**: active task selection, discrepancy-based transfer (single source)

Towards Lifelong Learning



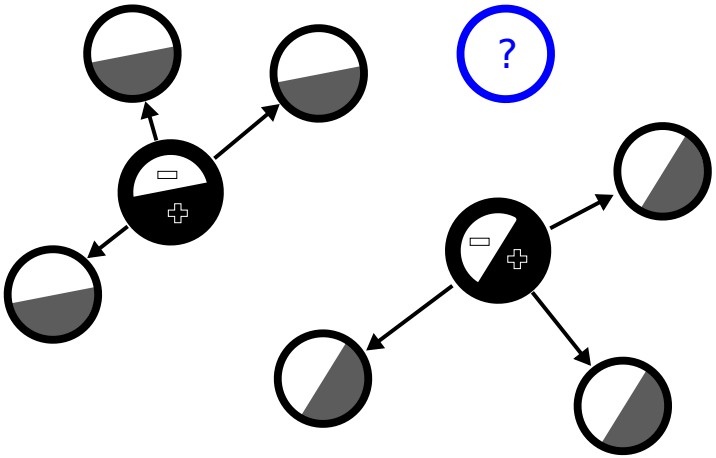
From Multi-Task to Lifelong Learning...

How to handle a new task?



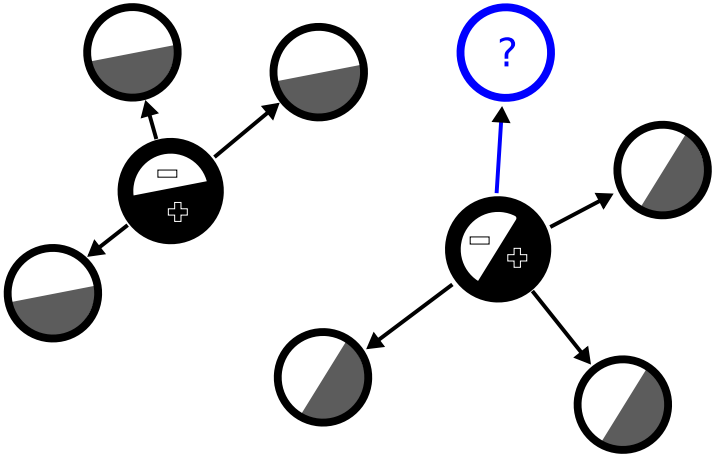
Given: previously learned tasks

How to handle a new task?



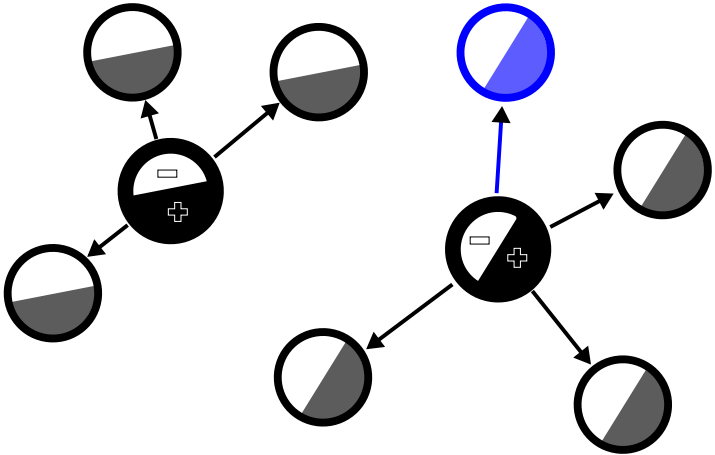
. . . a new task (unlabeled) needs to be solved

How to handle a new task?



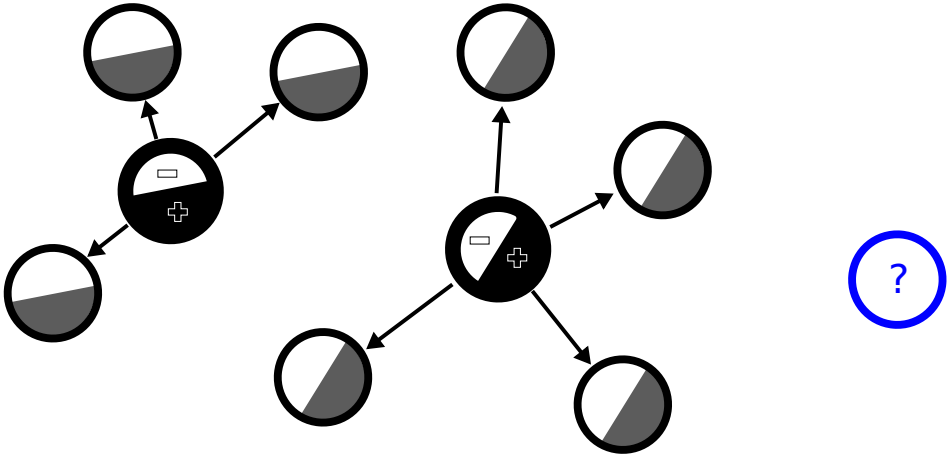
. . . measure discrepancy to closest labeled tasks

How to handle a new task?



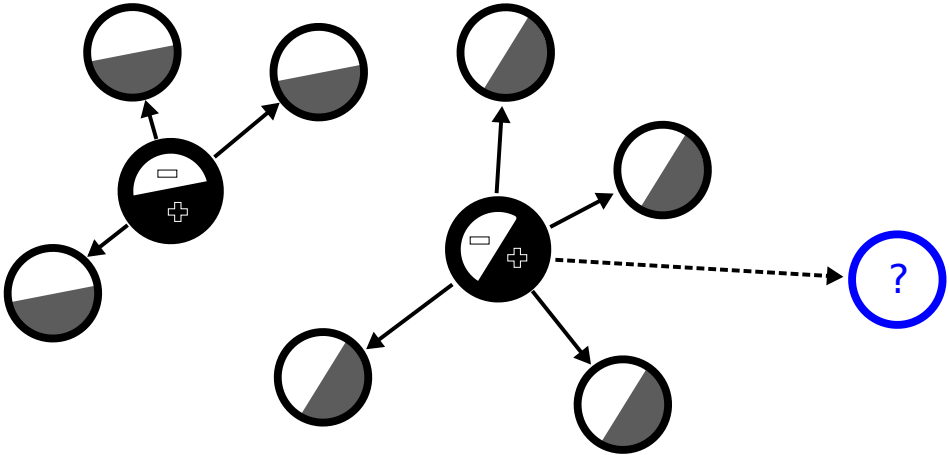
. . . if small enough, transfer classifier

How to handle a new task?



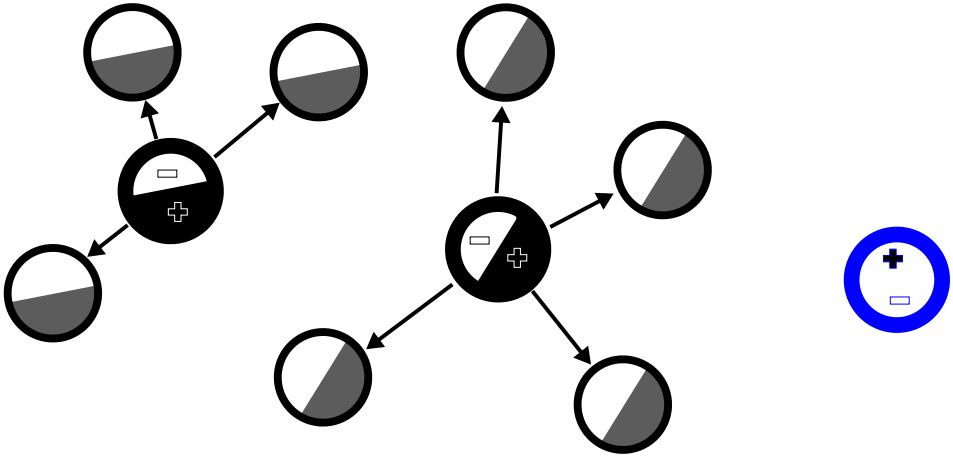
. . . a new task needs to be solved

How to handle a new task?



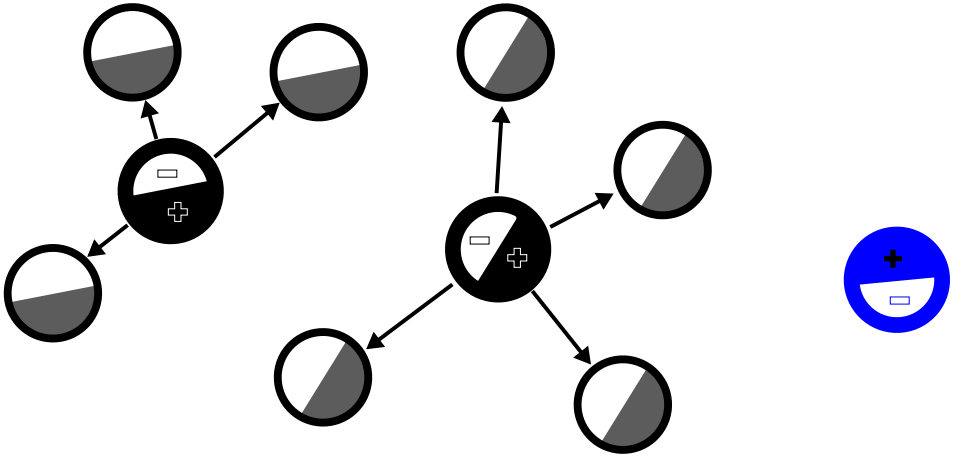
. . . measure discrepancy to closest labeled tasks

How to handle a new task?



. . . if too large, request labels

How to handle a new task?



. . . if too large, request labels and learn new classifier

Summary

Multi-Task Learning with Unlabeled Tasks

- many (similar) classification tasks to solve
- avoid having to collect annotation for all of them
- transfer information from labeled to unlabeled tasks
- principled algorithms, derived from generalization bounds

Active Task Selection

- identify most informative tasks and have only those labeled

Lifelong Learning

- new tasks coming in: use similarity to previous tasks to
  - solve using a previous classifier, or
  - ask for new training annotation

Thanks to...

My research group at IST Austria:


Alex Kolesnikov


Georg Martius


Asya Pentina


Amélie Royer


Alex Zimin

Funding Sources:

IST AUSTRIA
Institute of Science and Technology

erc

Reminder: Postdoc and PhD positions! → `http://www.ist.ac.at/~chl`